Bayesian anomaly detection for Cosmology - 21cm, Supernovae, and beyond

Sam Leeney

With: Will Handley, Eloy de Lera Acedo, Harry Bevins

May 27, 2025

Outline

Bayesian anomaly detection

Apply to 21cm Cosmology

Apply on Ia supernovae

What next

Bayesian anomaly detection

Over simplified example of anomaly detection method (thresholding)

Simple statistical approach:

- Calculate mean (μ) and standard deviation (σ) of the data
- Define threshold $T = \mu + k\sigma$ where k is a sensitivity parameter
- Flag data point x_i as anomalous if:

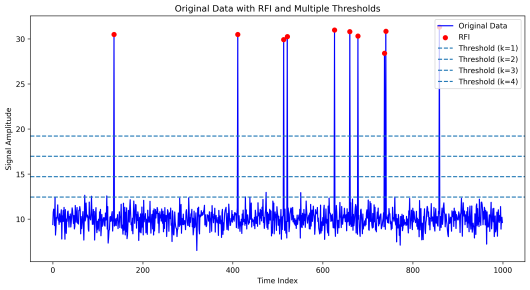
$$\text{anomaly}_i = \begin{cases} 1 & \text{if } x_i > T \\ 0 & \text{otherwise} \end{cases} \tag{1}$$

Limitations:

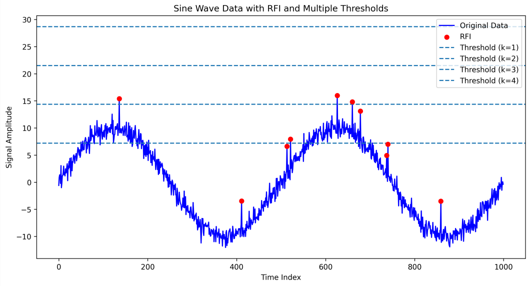
- Choice of k is arbitrary
- Assumes Gaussian statistics
- No consideration of temporal correlations
- Cannot distinguish between RFI and real signals

Thresholding results

Constant signal with RFI:



Sine wave with RFI:



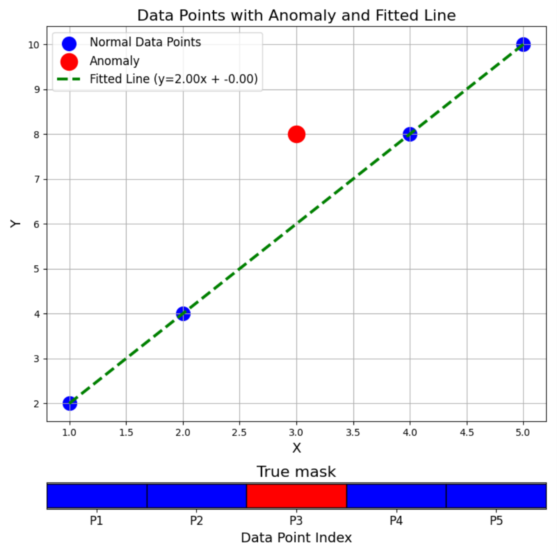
- Traditional methods are generally not model aware.
- Anomalies are typically sought either before or after typical fitting process.

Model anomalies in via a method that is:

- Model aware
- Works simultaneously with model fitting
- Not binary, ie encodes 'belief' datum are anomalous

Define anomaly mask ε

$$\varepsilon_i = \begin{cases} 0 & : \text{expected} \\ 1 & : \text{anomalous}, \end{cases} \tag{2}$$



Ascribe Bernoulli prior to ε

$$P(\varepsilon_i) = p^{\varepsilon_i}(1-p)^{(1-\varepsilon_i)}. \tag{3}$$

- A Bernoulli prior assigns a probability p to a binary variable being 1 (anomalous) and $1-p$ to it being 0 (expected).

Peicewise likelihood with ε

The likelihood function before marginalizing over ϵ is given by:

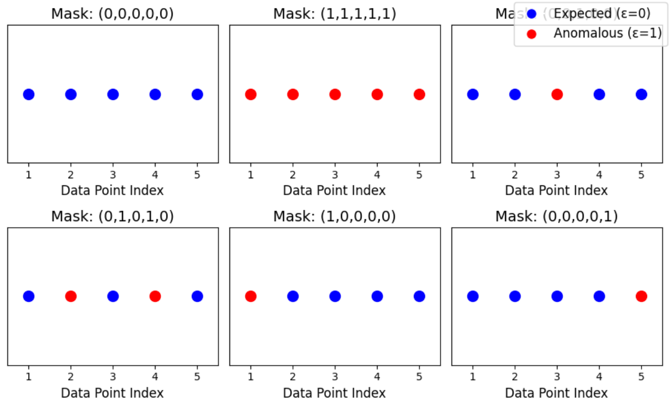
$$P(\vec{D}, \vec{\epsilon}|\theta) = \prod_{i=1}^{N} (L_i(\theta)(1-p))^{(1-\epsilon_i)} \left(\frac{p}{\Delta}\right)^{\epsilon_i}$$

Where:

- $L_i(\theta)$ is the likelihood of the i'th data point D_i under the "expected" model.
- Δ is a constant related to the "anomalous" model.
- p is the prior probability that a data point is anomalous ($P(\epsilon_i = 1)$).
- ϵ_i is a binary variable: $\epsilon_i = 0$ for expected, $\epsilon_i = 1$ for anomalous.

Marginalise over epsilon

$$P(\mathcal{D}|\theta) = \sum_{\varepsilon \in \{0,1\}^N} P(\mathcal{D}, \varepsilon|\theta) \tag{4}$$



Likelihood After Marginalization

The likelihood function after marginalizing over ϵ is given by:

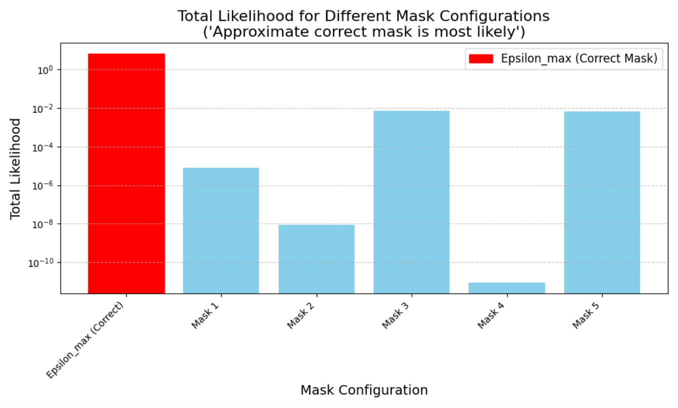
$$L(D|\theta) = \prod_{i=1}^{N} \left((1-p)L_i(\theta) + p\frac{1}{\Delta} \right)$$

Where:

- $D = \{D_1, D_2, \ldots, D_N\}$ represents the dataset of N data points.
- θ represents the model parameters.
- $L_i(\theta)$ is the likelihood of the i-th data point D_i being "expected".
- p is the prior probability that a single data point is "anomalous".
- This is computationally impractical as mask scales 2^N.

Approximate correct mask is most likely

$$P(\mathcal{D}|\theta, \varepsilon_{\max}) \gg \max_j P(\mathcal{D}|\theta, \varepsilon^{(j)}), \tag{5}$$



Loglikelihood and Maximisation

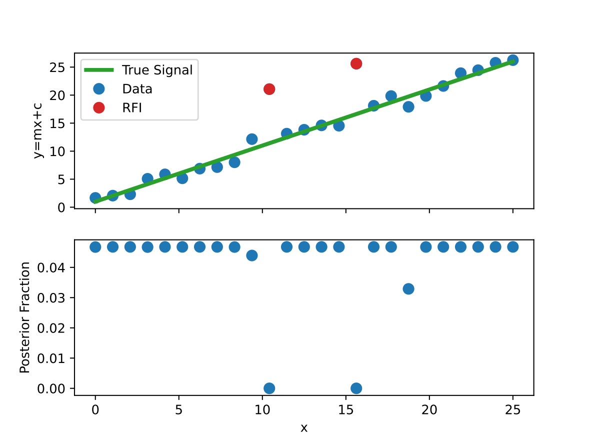
e) Loglikelihood:

$$\log P(\mathcal{D}|\theta) = \sum_i [\log \mathcal{L}_i + \log(1-p)]\varepsilon_i^{\mathrm{max}} + [\log p - \log \Delta](1-\varepsilon_i^{\mathrm{max}}) \tag{6}$$

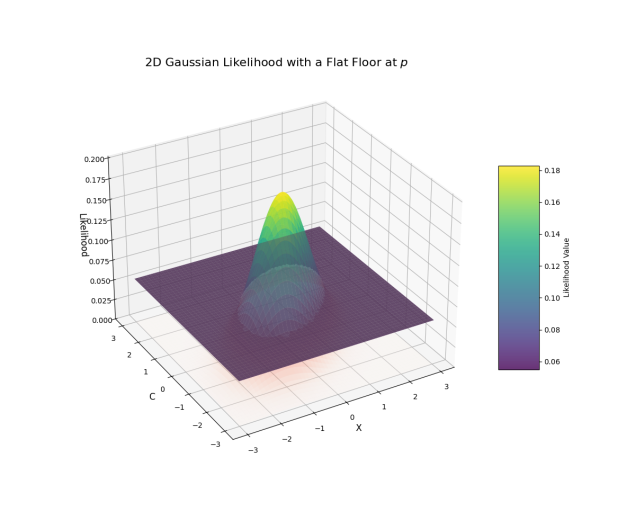
f) Find the mask that ε^{max} that maximises the likelihood by comparing the terms:

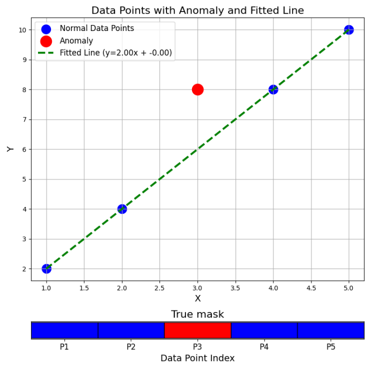
$$\log P(\mathcal{D}|\theta) = \begin{cases} \log \mathcal{L}_i + \log(1-p), & \text{if } [\log \mathcal{L}_i + \log(1-p) > \log p - \log \Delta] \\ \log p - \log \Delta, & \text{otherwise} \end{cases} \tag{7}$$

Fit on a simple toy model

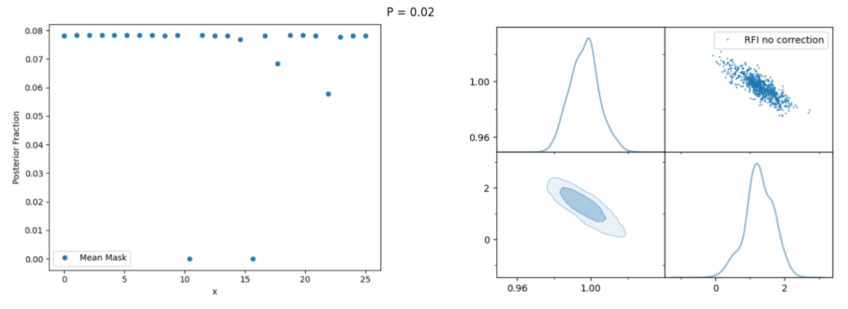


We are imposing a 'floor' on our likelihood

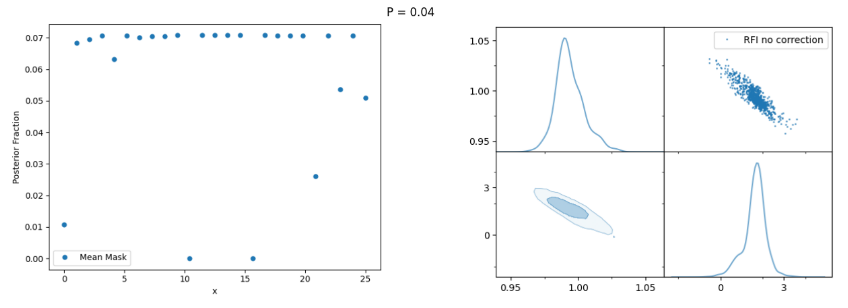




Varying p

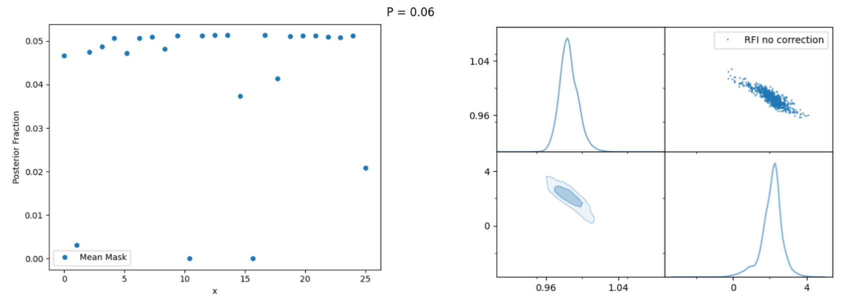

P = 0.02
Posterior Fraction
0.08
0.07
0.06
0.05
0.04
0.03
0.02
0.01
0.00
Mean Mask
0
5
10
15
20
25
x
RFI no correction
1.00
0.96
2
0
0.96
1.00
0
2

Varying p



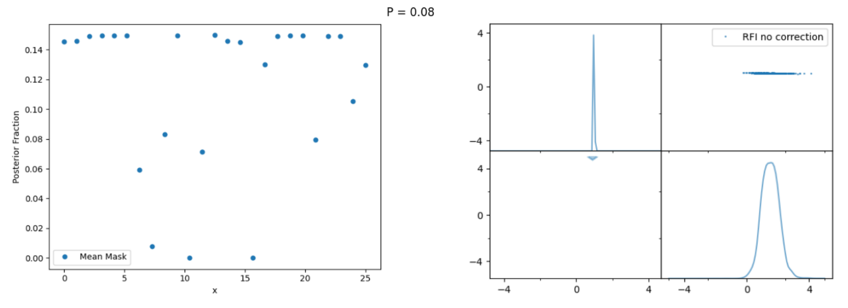
P = 0.04
Posterior Fraction
x
Mean Mask
0.00
0.01
0.02
0.03
0.04
0.05
0.06
0.07
0
5
10
15
20
25
RFI no correction
1.05
1.00
0.95
3
0
0.95
1.00
1.05
0
3

Varying p

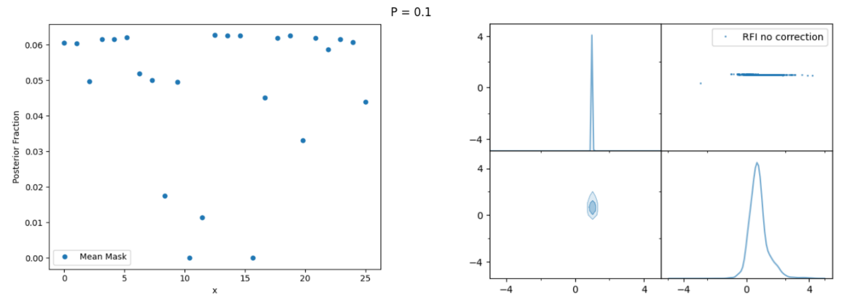


P = 0.06
Posterior Fraction
0.05
0.04
0.03
0.02
0.01
0.00
Mean Mask
0
5
10
15
20
25
x
RFI no correction
1.04
0.96
4
0
0.96
1.04
0
4

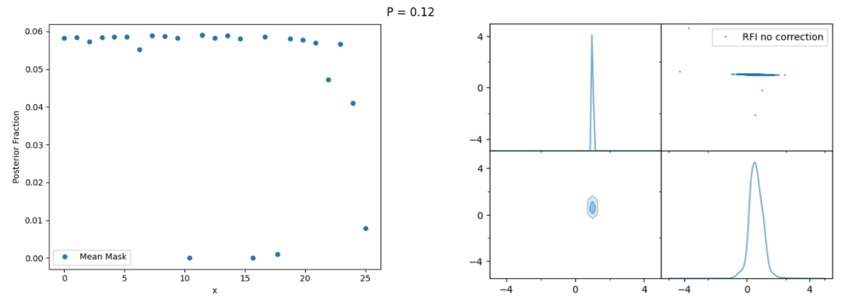
Varying p


P = 0.08
Posterior Fraction
0.00
0.02
0.04
0.06
0.08
0.10
0.12
0.14
x
0
5
10
15
20
25
Mean Mask
RFI no correction
4
0
−4
4
0
−4
−4
0
4
−4
0
4

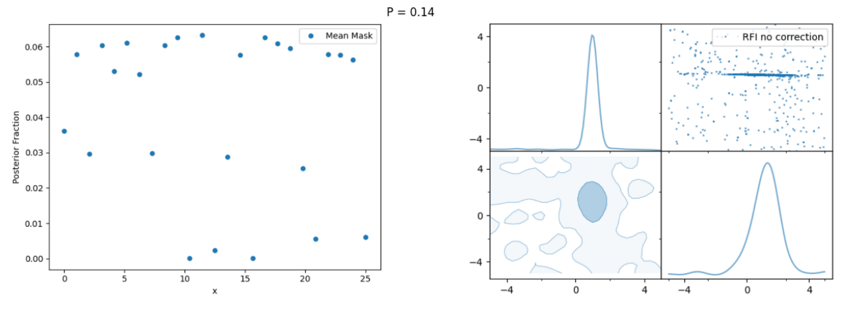
Varying p


P = 0.1
Posterior Fraction
0.06
0.05
0.04
0.03
0.02
0.01
0.00
Mean Mask
0
5
10
15
20
25
x
RFI no correction
4
0
−4
4
0
−4
−4
0
4
−4
0
4

Varying p

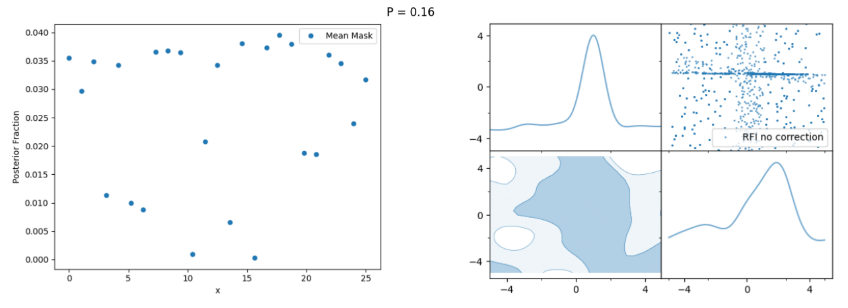


Varying p



P = 0.14
Mean Mask
Posterior Fraction
0.06
0.05
0.04
0.03
0.02
0.01
0.00
0
5
10
15
20
25
x
RFI no correction
4
0
−4
−4
0
4

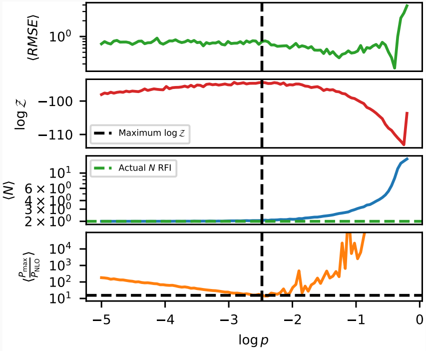
Varying p



P = 0.16
Mean Mask
Posterior Fraction
x
0.040
0.035
0.030
0.025
0.020
0.015
0.010
0.005
0.000
0
5
10
15
20
25
RFI no correction

Selection strategy for p.

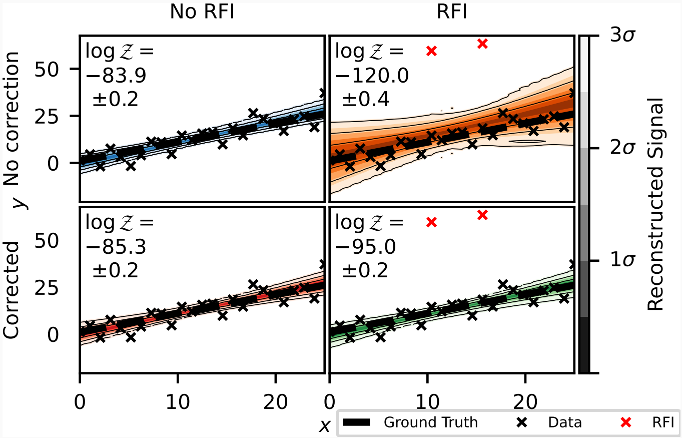
- 'Select p such that the Bayesian evidence is maximised'



Fully automated anomaly detection

- Putting a prior on p, we can fit it dynamically as a free parameter.
- This fully automates the anomaly detection process.
- Must exclude $p = 0$.

Application to toy model



Implement with 2 lines of code

```python
def likelihood(theta):
    sig = theta[0]
    logL = -(f_noise - window)**2/sig**2/2 - np.log(2*np.pi*sig**2)/2
    return logL, []
```

```python
def likelihood(theta):
    sig = theta[0]
    logL = -(f_noise - window)**2/sig**2/2 - np.log(2*np.pi*sig**2)/2 + np.log(1-p)
    emax = logL > logp - np.log(delta)
    logPmax = np.where(emax, logL, logp - np.log(delta)).sum()
    return logPmax, []
```

Tutorial @ github.com/samleeney

Read the paper!

PHYSICAL REVIEW D **108,** 062006 (2023)

Bayesian approach to radio frequency interference mitigation

S. A. K. Leeney,[*,§] W. J. Handley,[†,§] and E. de Lera Acedo[‡,§]
The University of Cambridge, Astrophysics Group, Cavendish Laboratory,
J. J. Thomson Avenue, Cambridge, CB3 0HE, United Kingdom

(Received 5 May 2023; accepted 29 August 2023; published 29 September 2023)

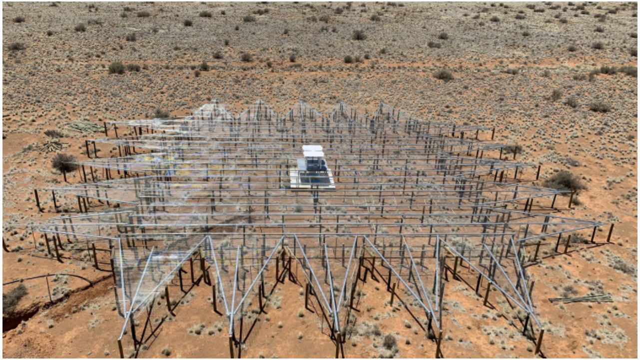
Interfering signals such as radio frequency interference from ubiquitous satellite constellations are becoming an endemic problem in fields involving physical observations of the electromagnetic spectrum. To address this we propose a novel data cleaning methodology. Contamination is simultaneously flagged and managed at the likelihood level. It is modeled in a Bayesian fashion through a piecewise likelihood that is constrained by a Bernoulli prior distribution. The techniques described in this paper can be implemented with just a few lines of code.

DOI: 10.1103/PhysRevD.108.062006

arxiv: 2211.15448

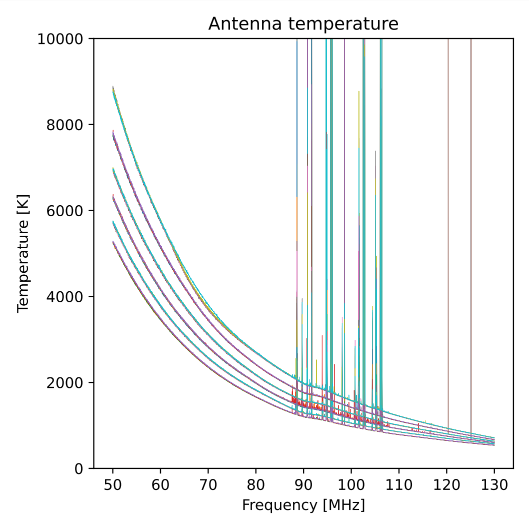
Apply to 21cm Cosmology

What is REACH?



- The redshifted sky-averaged 21cm line of neutral hydrogen carries the imprint of the first stars and galaxies formed during the Cosmic Dawn and Epoch of Reionisation
- 10^-5 times dimmer than foregrounds.
- Detection enables inference of fundamental physics and cosmology.

Heavily contaminated



Fitting a contaminated global 21cm signal

Standard Likelihood:

$$\log \mathcal{L} = \sum_i -\frac{1}{2} \log \left(2\pi\sigma_n^2\right) - \frac{1}{2} \left(\frac{T_{\mathrm{data}}(\nu_i) - (T_{\mathrm{model}}(\nu_i) + T_{21}(\nu_i))}{\sigma_n} \right)^2 .$$

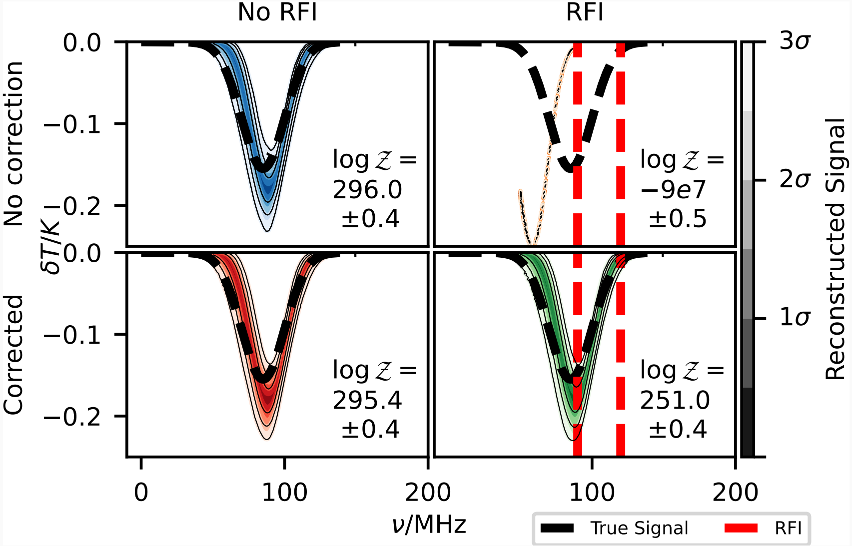
- $T_{\mathrm{data}}(\nu_i)$: Observed data at frequency ν_i
- $T_{\mathrm{model}}(\nu_i)$: Model for foregrounds and nuisance parameters
- $T_{21}(\nu_i)$: Global 21cm signal (signal of interest)
- σ_n: Noise uncertainty

Anomaly Detection Likelihood:

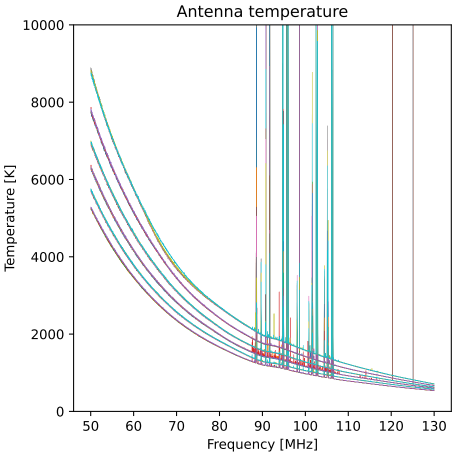
$$\log \mathcal{L}_{\mathrm{anom}} = \sum_i \begin{cases} \log \mathcal{L}_i + \log(1-p), & \text{if } e_i^{\mathrm{max}} \\ \log p - \log \Delta, & \text{otherwise} \end{cases}$$

- $\log \mathcal{L}_i$: Point-wise standard likelihood
- p: Anomaly probability (model parameter)
- e_i^{max}: Boolean indicating normal data
- Δ: Maximum value of the data range

Fitting a contaminated global 21cm signal



Speeding up...



- We fit many observations simultaneously.
- This gets very slow, so we need to speed up.

Speeding up...

RAS Techniques and Instruments

RASTAI **3,** 372–384 (2024) https://doi.org/10.1093/rasti/rzae025

Advance Access publication 2024 July 5

Enhanced Bayesian RFI mitigation and transient flagging using likelihood reweighting

Dominic Anstey [1,2★] and Samuel A. K. Leeney [1,2]

[1] *Astrophysics Group, Cavendish Laboratory, J. J. Thomson Avenue, Cambridge CB3 0HE, UK*

[2] *Kavli Institute for Cosmology, Madingley Road, Cambridge CB3 0HA, UK*

Accepted 2024 July 3. Received 2024 May 21; in original form 2023 October 12

Apply on Ia supernovae

JAX-bandflux: A Tool for Supernovae Analysis

Search projects

Help Sponsors Log in Register

jax-bandflux 0.3.1

```
pip install jax-bandflux
```

Latest version

Released: Apr 10, 2025

JAX-bandflux: differentiable supernovae SALT modelling for cosmological analysis on GPUs

Samuel Alan Kossoff Leeney[1, 2]

1 Astrophysics Group, Cavendish Laboratory, J. J. Thomson Avenue, Cambridge CB3 0HE, UK **2** Kavli Institute for Cosmology, Madingley Road, Cambridge CB3 0HA, UK

JAX-bandflux: A Tool for Supernovae Analysis

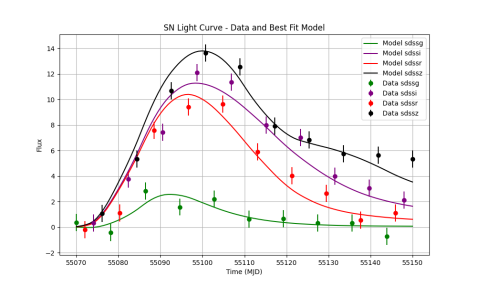
JAX-bandflux: differentiable supernovae SALT modelling for cosmological analysis on GPUs

Samuel Alan Kossoff Leeney[1, 2]

1 Astrophysics Group, Cavendish Laboratory, J. J. Thomson Avenue, Cambridge CB3 0HE, UK **2** Kavli Institute for Cosmology, Madingley Road, Cambridge CB3 0HA, UK

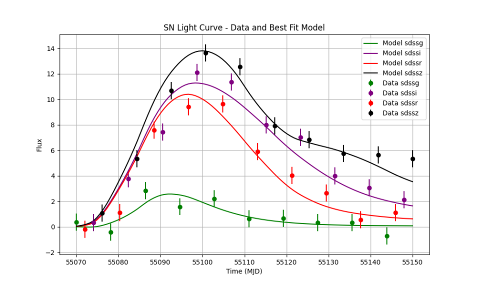
Fitting Supernovae Light Curves with `JAX-bandflux`

- Supernovae light curves measure the brightness of a supernova over time across different wavelengths.
- `JAX-bandflux` is a Python package for fitting supernova light curves.
- It provides tools for model definition, fitting, and simulation.
- GPU acceleration significantly speeds up the fitting process for large datasets.



The SALT Model for Supernovae

- Widely used empirical model for Type Ia supernovae light curves.
- Describes supernova flux as a function of wavelength and time.



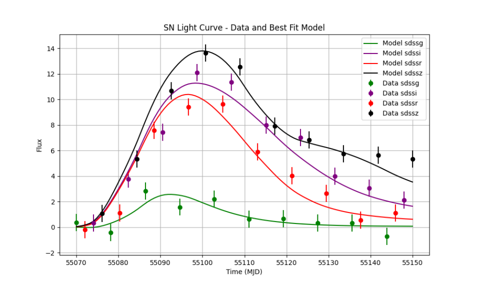
Model flux:

$$F(p,\lambda) = x_0\left[M_0(p,\lambda) + x_1 M_1(p,\lambda) + \ldots\right] \times \exp\left[c \times CL(\lambda)\right]$$

Bandflux Computation

- Observed flux integrated over a specific photometric bandpass.
- Calculated by convolving SED with instrument's bandpass transmission.
- Formula:

$$\text{bandflux} = \int_{\lambda_{\text{min}}}^{\lambda_{\text{max}}} F(\lambda) \cdot T(\lambda) \cdot \frac{\lambda}{hc} d\lambda$$



Cosmology from SALT parameters

- SALT parameters (x_0, x_1, c) are fitted to Type Ia supernovae light curves.
- These parameters standardize SNe Ia into standard candles by correcting for luminosity, shape (x_1), and color (c).
- This standardization allows for precise calculation of distance moduli, linking to luminosity distance.
- By fitting cosmological models to a large sample of these calibrated supernova distances, cosmological parameters, such as the Hubble constant (H_0), can be determined.

Likelihood Function for Supernovae Fitting

- The likelihood function quantifies how well a given model (e.g., SALT) explains the observed data.
- For photometric observations, assuming Gaussian uncertainties, the likelihood is typically defined as:

$$\mathcal{L}(\theta) = \prod_{i=1}^{N} \frac{1}{\sqrt{2\pi\sigma_i^2}} \exp\left(-\frac{(f_i^{\text{obs}} - f_i^{\text{model}}(\theta))^2}{2\sigma_i^2}\right)$$

  where f_i^{obs} are observed fluxes, $f_i^{\text{model}}(\theta)$ are model fluxes, and σ_i are uncertainties.
- In a Bayesian context, this likelihood is combined with priors on the model parameters to form the posterior distribution.
- GPU compatibility allows for rapid evaluation of the likelihood across a large parameter space, enabling efficient sampling and inference.

Standard vs. Anomaly Detection Likelihoods

Standard Likelihood:

$$\log \mathcal{L}_{\text{std}} = -\frac{1}{2}\sum_i \left(\frac{f_i - m_i}{\sigma_i}\right)^2 - \frac{1}{2}\sum_i \log(2\pi\sigma_i^2) \qquad (8)$$

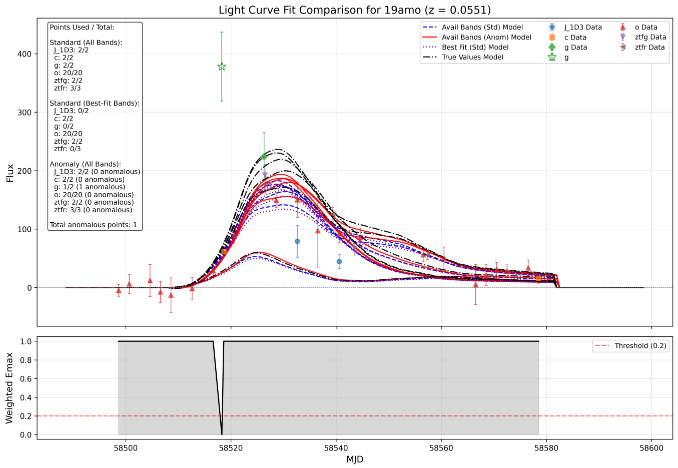
- f_i: Observed flux
- m_i: Model flux (SALT3)
- σ_i: Flux uncertainty

Anomaly Detection Likelihood:

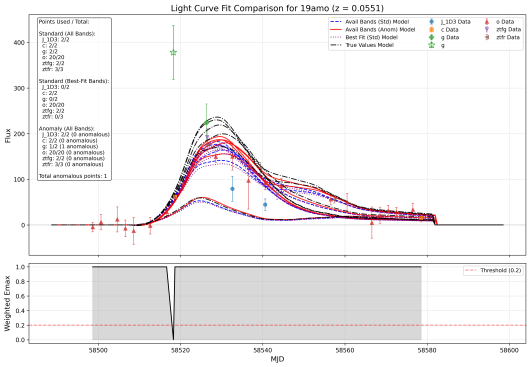
$$\log \mathcal{L}_{\text{anom}} = \sum_i \begin{cases} \log \mathcal{L}_i + \log(1-p), & \text{if } e_i^{\text{max}} \\ \log p - \log \Delta, & \text{otherwi} \end{cases} \qquad (9)$$

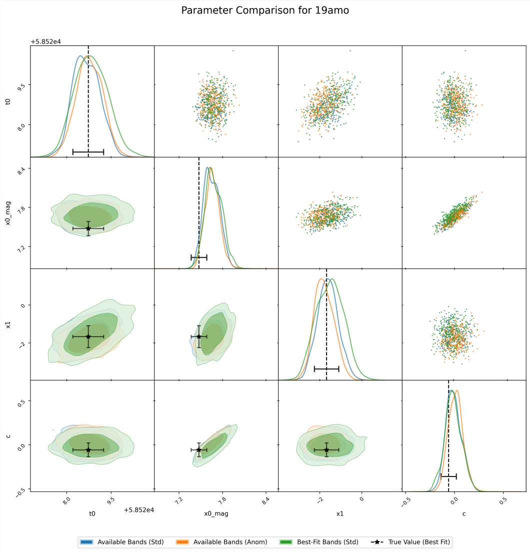
- $\log \mathcal{L}_i$: Point-wise standard likelihood
- p: Anomaly probability (fitted parameter)
- e_i^{max}: Boolean indicating normal data
- Δ: Maximum flux range

Applying to Ia supernovae

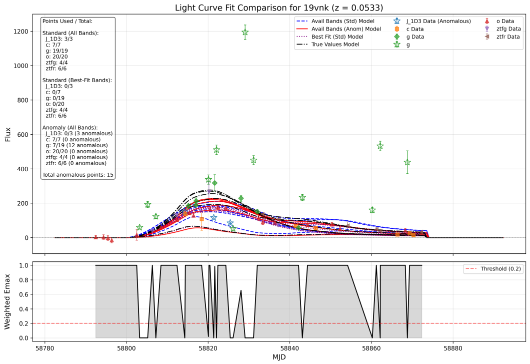


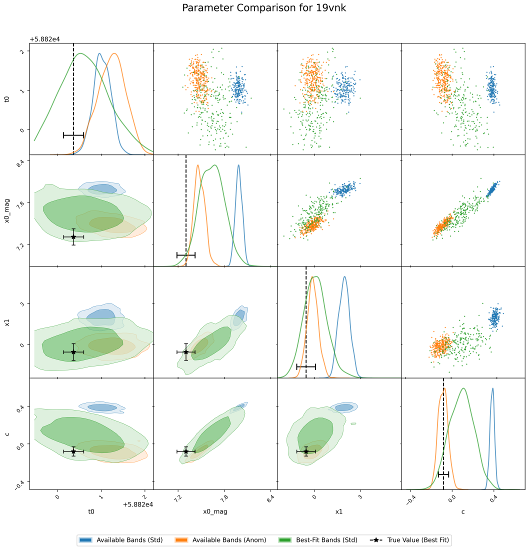
SN 19amo: Classic 'anomaly detection' example



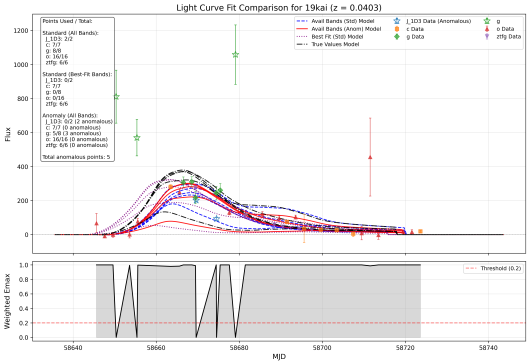


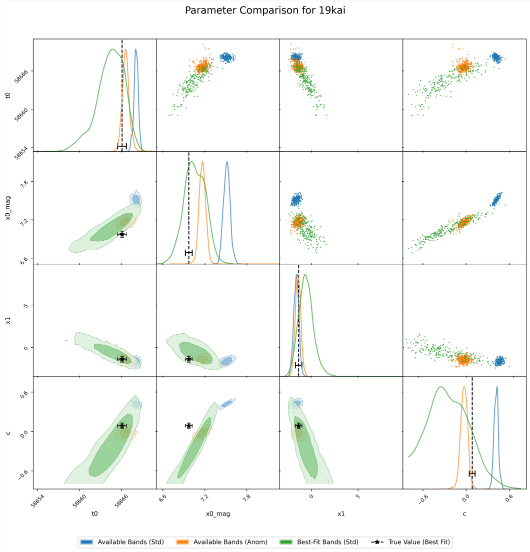
SN 19vnk: Automatic filter removal



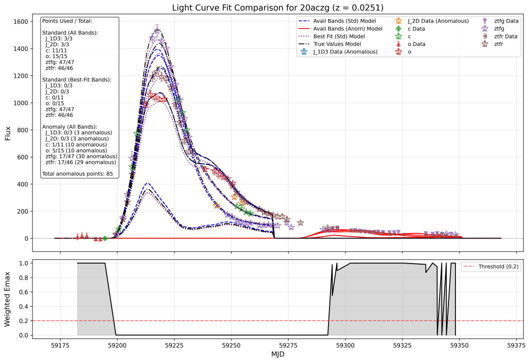


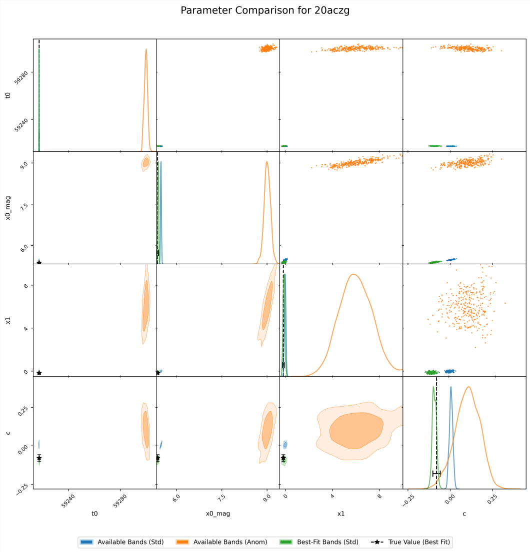
SN 19kai: Flagging while preserving some data





SN 20aczg: Light Curve and Corner Plot Comparison





Key points

1. Standard flagging.

Key points

1. Standard flagging.
2. Automated filter selection.

Key points

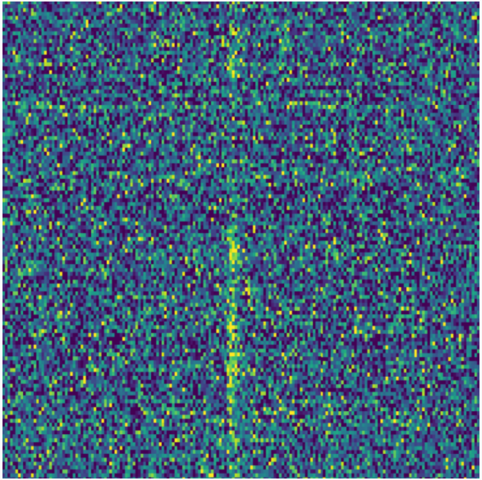
1. Standard flagging.
2. Automated filter selection.
3. Data preservation from previously discarded filters.
4. Potentially can flag non Ia automatically?

Next steps

- Assess Hubble diagrams
  - Quantify impact on cosmological parameter estimation
  - Compare with traditional outlier rejection methods
  - Evaluate systematic error reduction
- Try on other datasets?
  - Apply to different supernova surveys (ZTF, LSST)
  - Test with different photometric systems
  - Evaluate performance across redshift ranges

What next

What do(nt) we look for?



- New science often found when looking for something else.
- How to search for something unknown?
- How much new science was missed in old data?